Michio MASUDA et al.
"Multi-Layer Class Identifying…"
Q62568
Filed January 3, 2001
Sheets 1 of 16

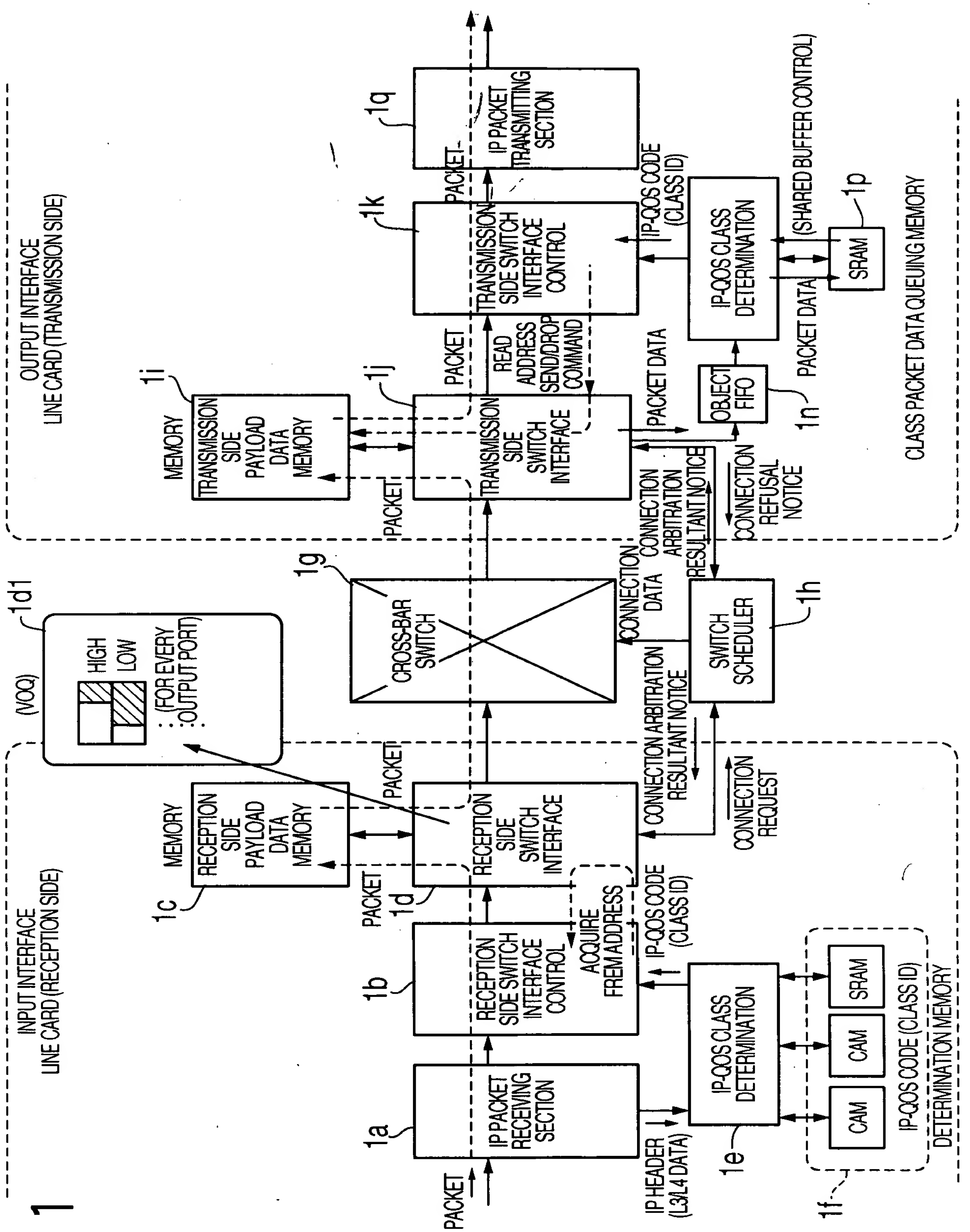


Fig. 1

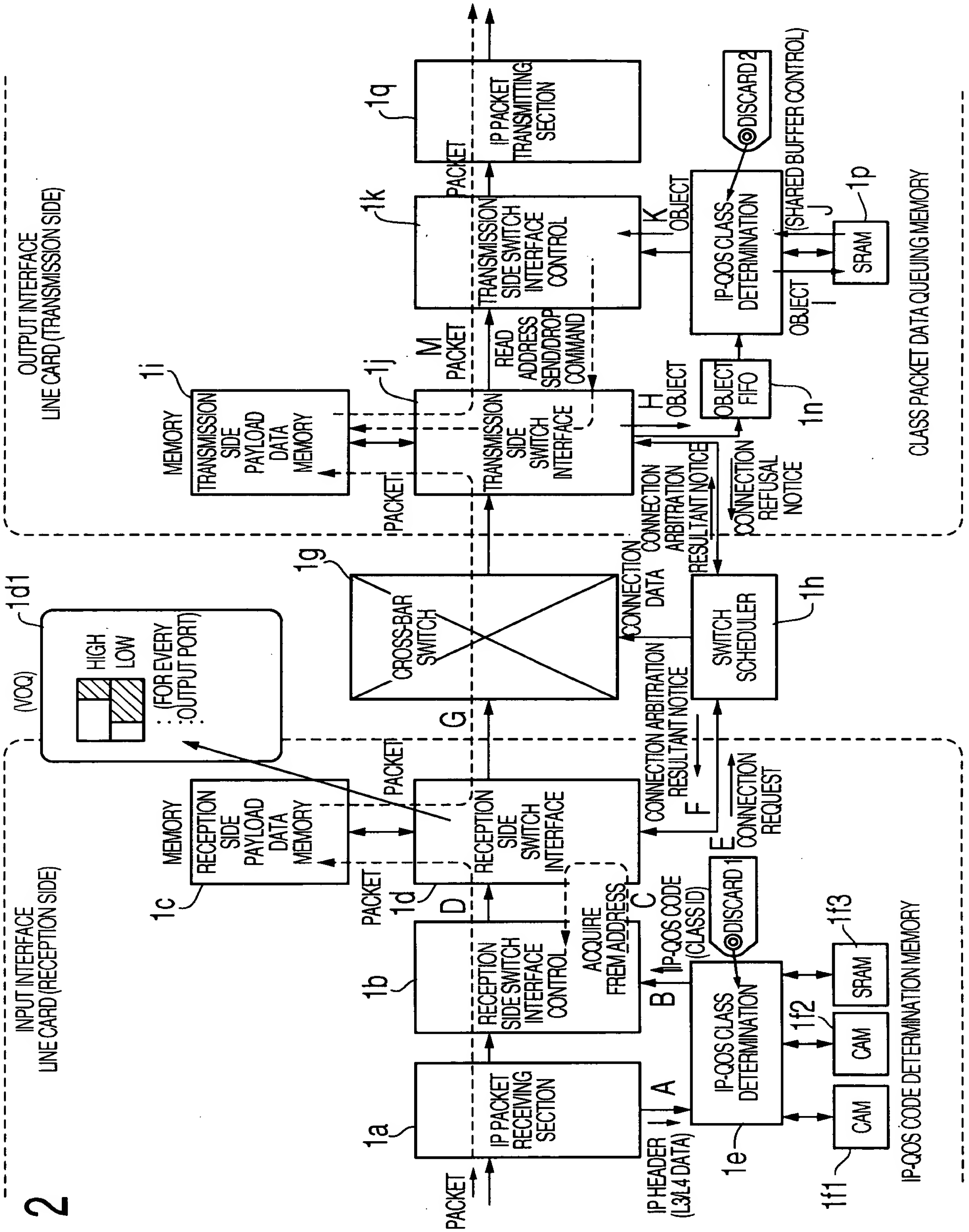

INPUT INTERFACE
LINE CARD (RECEPTION SIDE)
OUTPUT INTERFACE
LINE CARD (TRANSMISSION SIDE)
PACKET
1a
IP PACKET RECEIVING SECTION
1b
RECEPTION SIDE SWITCH INTERFACE CONTROL
ACQUIRE FREM ADDRESS
1c
MEMORY
RECEPTION SIDE PAYLOAD DATA MEMORY
PACKET
1d
D
RECEPTION SIDE SWITCH INTERFACE
PACKET
(VOQ)
1d1
HIGH
LOW
(FOR EVERY OUTPUT PORT)
1g
CROSS-BAR SWITCH
G
CONNECTION DATA
CONNECTION ARBITRATION RESULTANT NOTICE
F
E
CONNECTION REQUEST
1h
SWITCH SCHEDULER
CONNECTION ARBITRATION RESULTANT NOTICE
CONNECTION REFUSAL NOTICE
1i
MEMORY
TRANSMISSION SIDE PAYLOAD DATA MEMORY
PACKET
1j
TRANSMISSION SIDE SWITCH INTERFACE
M
PACKET
READ ADDRESS SEND/DROP COMMAND
H
OBJECT
1k
TRANSMISSION SIDE SWITCH INTERFACE CONTROL
K
OBJECT
PACKET
1q
IP PACKET TRANSMITTING SECTION
IP HEADER (L3/L4 DATA)
A
B
IP-QOS CODE (CLASS ID)
C
1e
IP-QOS CLASS DETERMINATION
DISCARD 1
1f1
CAM
1f2
CAM
1f3
SRAM
IP-QOS CODE DETERMINATION MEMORY
1n
OBJECT FIFO
OBJECT
I
IP-QOS CLASS DETERMINATION
DISCARD 2
J
(SHARED BUFFER CONTROL)
1p
SRAM
CLASS PACKET DATA QUEUING MEMORY

Fig. 2

Fig. 3

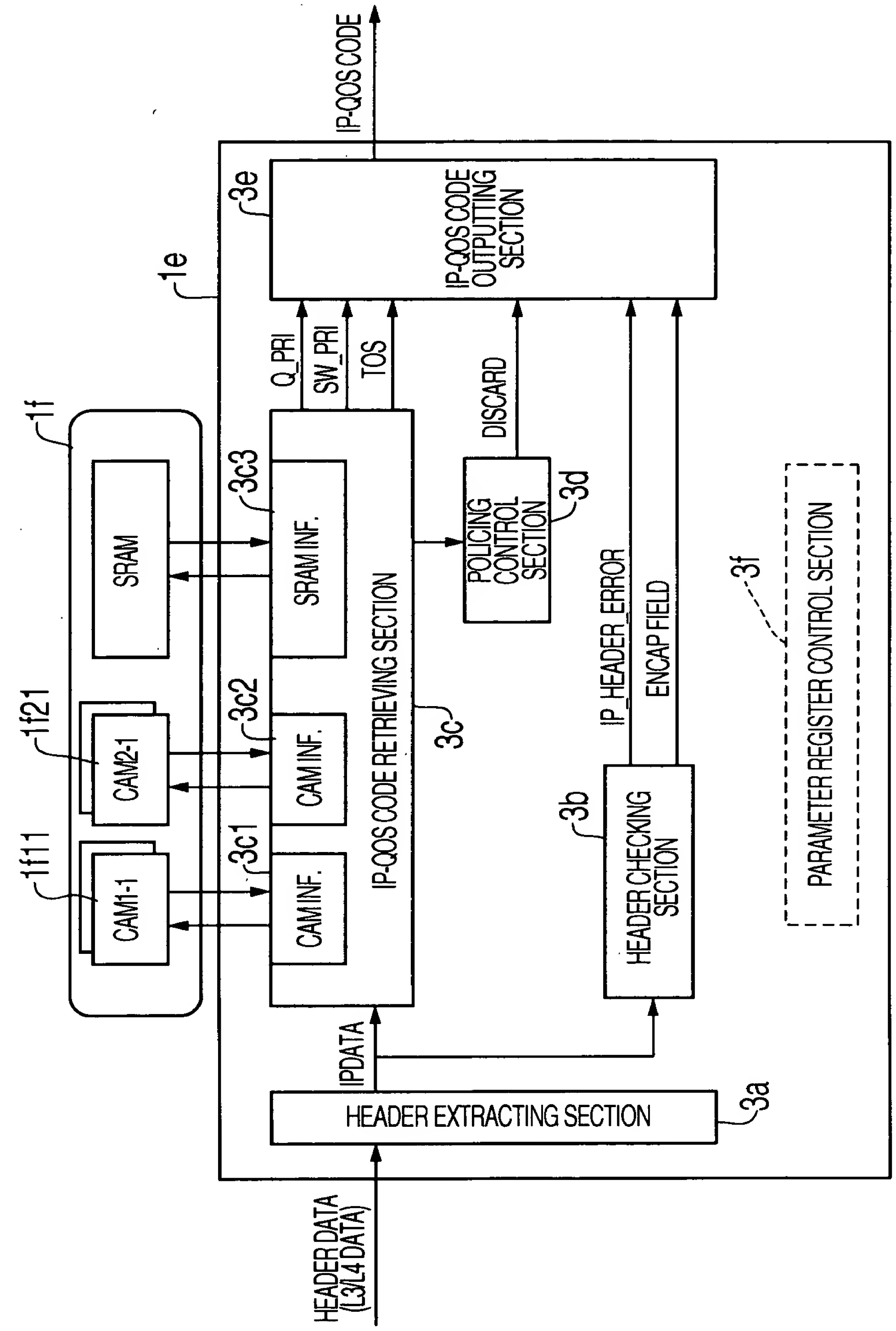
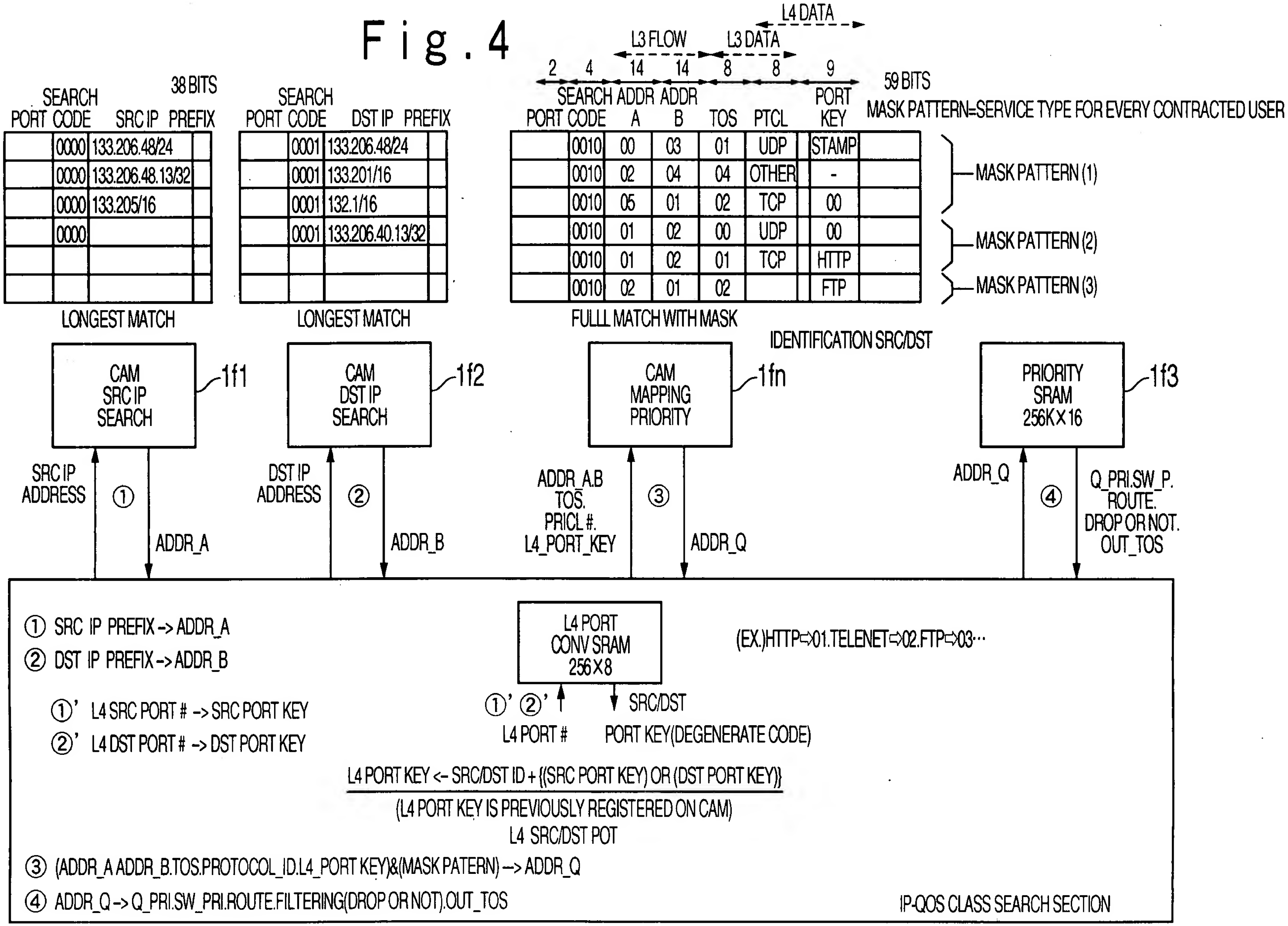

HEADER DATA (L3/L4 DATA)
3a
HEADER EXTRACTING SECTION
IPDATA
1e
3b
HEADER CHECKING SECTION
IP_HEADER_ERROR
ENCAP FIELD
3c
IP-QOS CODE RETRIEVING SECTION
3c1
CAM INF.
3c2
CAM INF.
3c3
SRAM INF.
1f
1f11
CAM1-1
1f21
CAM2-1
SRAM
3d
POLICING CONTROL SECTION
DISCARD
Q_PRI
SW_PRI
TOS
3e
IP-QOS CODE OUTPUTTING SECTION
IP-QOS CODE
3f
PARAMETER REGISTER CONTROL SECTION

Fig. 4



SEARCH PORT	CODE	SRC IP PREFIX	
	0000	133.206.48/24	
	0000	133.206.48.13/32	
	0000	133.205/16	
	0000		

SEARCH PORT	CODE	DST IP PREFIX	
	0001	133.206.48/24	
	0001	133.201/16	
	0001	132.1/16	
	0001	133.206.40.13/32	

SEARCH PORT	CODE	ADDR A	ADDR B	TOS	PTCL	PORT KEY	
	0010	00	03	01	UDP	STAMP	
	0010	02	04	04	OTHER	-	
	0010	05	01	02	TCP	00	
	0010	01	02	00	UDP	00	
	0010	01	02	01	TCP	HTTP	
	0010	02	01	02		FTP	

① SRC IP PREFIX -> ADDR_A
② DST IP PREFIX -> ADDR_B

①' L4 SRC PORT # -> SRC PORT KEY
②' L4 DST PORT # -> DST PORT KEY

L4 PORT KEY <- SRC/DST ID + {(SRC PORT KEY) OR (DST PORT KEY)}
(L4 PORT KEY IS PREVIOUSLY REGISTERED ON CAM)
L4 SRC/DST POT

③ (ADDR_A ADDR_B.TOS.PROTOCOL_ID.L4_PORT KEY)&(MASK PATERN) --> ADDR_Q
④ ADDR_Q -> Q_PRI.SW_PRI.ROUTE.FILTERING(DROP OR NOT).OUT_TOS

Fig.5

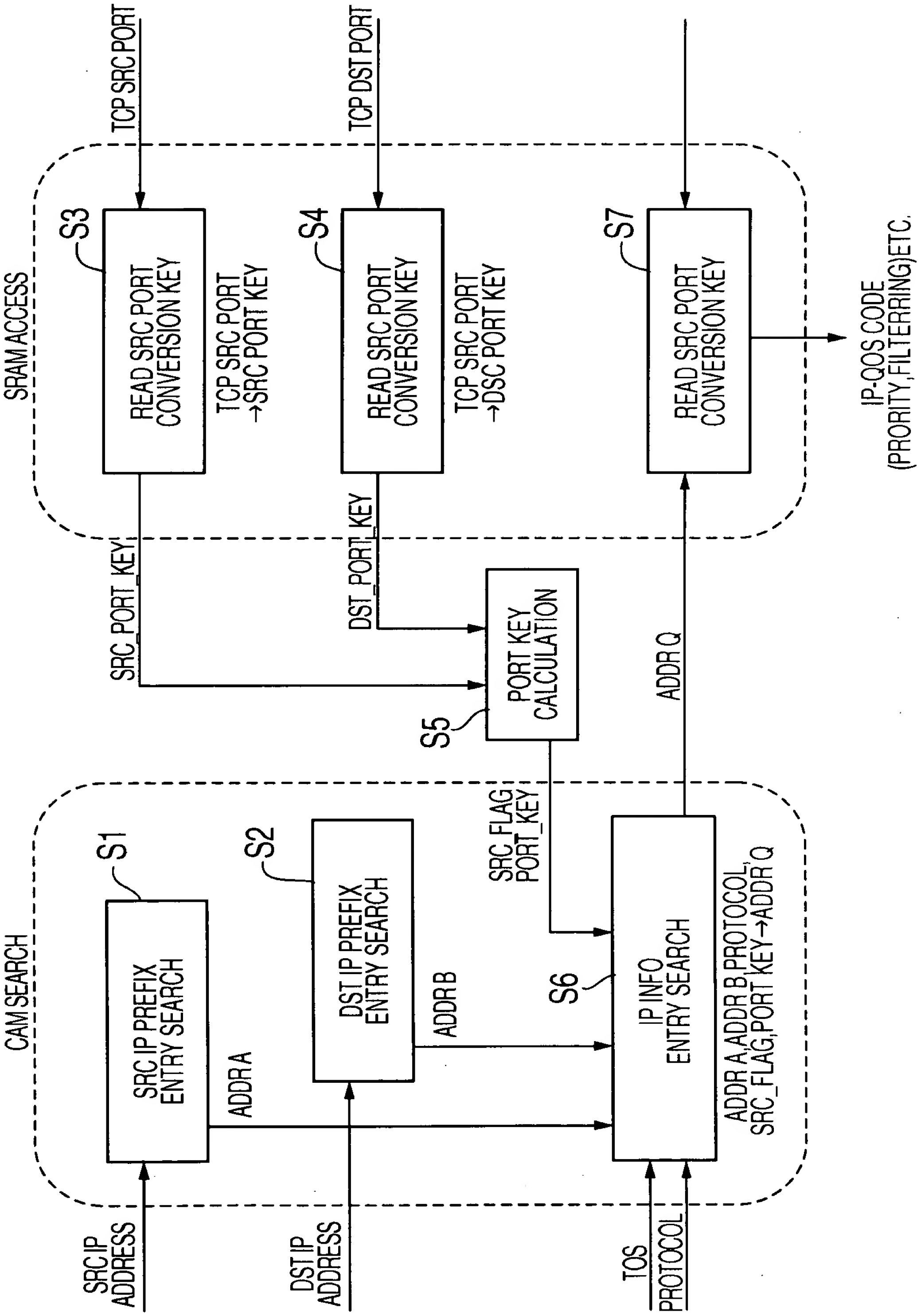

SRC IP ADDRESS
DST IP ADDRESS
TOS
PROTOCOL
CAM SEARCH
S1
SRC IP PREFIX ENTRY SEARCH
ADDR A
S2
DST IP PREFIX ENTRY SEARCH
ADDR B
S6
IP INFO ENTRY SEARCH
ADDR A,ADDR B,PROTOCOL, SRC_FLAG,PORT KEY→ADDR Q
SRC_FLAG PORT_KEY
S5
PORT KEY CALCULATION
SRC_PORT_KEY
DST_PORT_KEY
ADDR Q
SRAM ACCESS
TCP SRC PORT
S3
READ SRC PORT CONVERSION KEY
TCP SRC PORT →SRC PORT KEY
TCP DST PORT
S4
READ SRC PORT CONVERSION KEY
TCP SRC PORT →DSC PORT KEY
S7
READ SRC PORT CONVERSION KEY
IP-QOS CODE (PRORITY,FILTERRING)ETC.

Fig. 6A

【CAM REGION DIVISION】

CAM ADDRESS		CAM DATA (MAX.64 BITS)	MASK PATTERN (64 BITS)	SEARCH METHOD
	ADDR_A~	IP SRC PREFIX ENTRY STORAGE REGION		LONGEST MATCH
	ADDR_B~	IP DST PREFIX ENTRY STORAGE REGION		LONGEST MATCH
	ADDR_Q~	IP INFO SEARCH ENTRY STORAGE REGION		FULL MATCH WITH MASK

Fig. 6B

【1,IP SRC PREFIX ENTRY STORAGE REGION : SEARCH CODE 0000】

CAM ADDRESS (ADDR_A)	CAM DATA (38 BITS)			
	HW #(2)	SEARCH CODE(4)	IP SRC ADRESS/ PREFIX(32BITS)	NON USED(26 BITS)
A #1	00	0000	IP SRC ADDRESS #1/PREFIX	
A #2	00	0000	IP SRC ADDRESS #2/PREFIX	
A #3	01	0000	IP SRC ADDRESS #1/PREFIX	
⋮	⋮	⋮	⋮	

Fig.7A

【2,IP DST PREFIX ENTRY STORAGE REGION : SEARCH CODE 0001】

CAM ADDRESS (ADDR_B)	CAM DATA (38 BITS)			
	HW #(2)	SEARCH CODE(4)	IP DST ADDRESS/ PREFIX(32BITS)	NON USED(26 BITS)
B #1	00	0001	IP DST ADDRESS #1/PREFIX	
B #2	00	0001	IP DST ADDRESS #2/PREFIX	
B #3	01	0001	IP DST ADDRESS #1/PREFIX	
⋮	⋮	⋮	⋮	

Fig.7B

【3,IP INFO ADDRESS ENTRY STORAGE REGION : SEARCH CODE 0010】

CAM ADDRESS (ADDR_Q)	CAM DATA (55 BITS)								
	HW #(2)	SEARCH CODE(4)	ADDR A(14)	ADDR B(14)	TOS (8)	PROTO COL# (8)	SRC/ DST (1)	PORT KEY (8)	NON USED (5 BITS)
Q #1	00	0010	A1	B1	01	TCP	S	HTTP	
Q #2	00	0010	A1	B2	04	UDP	D	SNMP	
Q #3	01	0010	A3	B1	02	TCP	S	FTP	
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	

Fig.8

【IP INFO ENTRY】

ADDRESS(16 BITS) : UPPER 2 BITS=00 LOWER 14 BITS=HIT ADDR_Q	DATA(24 BIT)							
	Q_PRI(4)	D	P	ROUTE(1+4)		OUTPUT TOS(2+8)		RESERVE (3)
ADDR Q0	0000	0	0	0	0000	11	011011 00	
ADDR Q1	1101	0	1	0	0000	11	011010 00	
ADDR Q2	1101	0	0	0	0000	00	000000 00	
⋮	⋮	⋮	⋮		⋮		⋮	
ADDR QI	1110	0	1	1	0101	00	000000 00	
⋮	⋮	⋮	⋮		⋮		⋮	

Fig. 9

(IPV4 & TCP/UDP/OTHER HEADER FORMAT)

<table>
<tr><td>WORD</td><td colspan="3">63</td><td>47</td><td>31</td><td colspan="3">15</td></tr>
<tr><td>-</td><td colspan="4">EMPTY DATA</td><td colspan="4">PPP HEADER</td></tr>
<tr><td>0</td><td>VER</td><td>IHL</td><td>TOS</td><td>DETAGRAM LENGTH</td><td>IDENTIFICATION</td><td></td><td>M</td><td>FRAGMENT OFFSET</td></tr>
<tr><td>1</td><td>TTL</td><td></td><td>PROTOCOL</td><td>HEADER CHECKSUM</td><td colspan="4">SRC IP ADDRESS</td></tr>
<tr><td>2</td><td colspan="4">DST IP ADDRESS</td><td>L4 SRC PORT</td><td colspan="3">L4 DST PORT</td></tr>
</table>

Fig.10

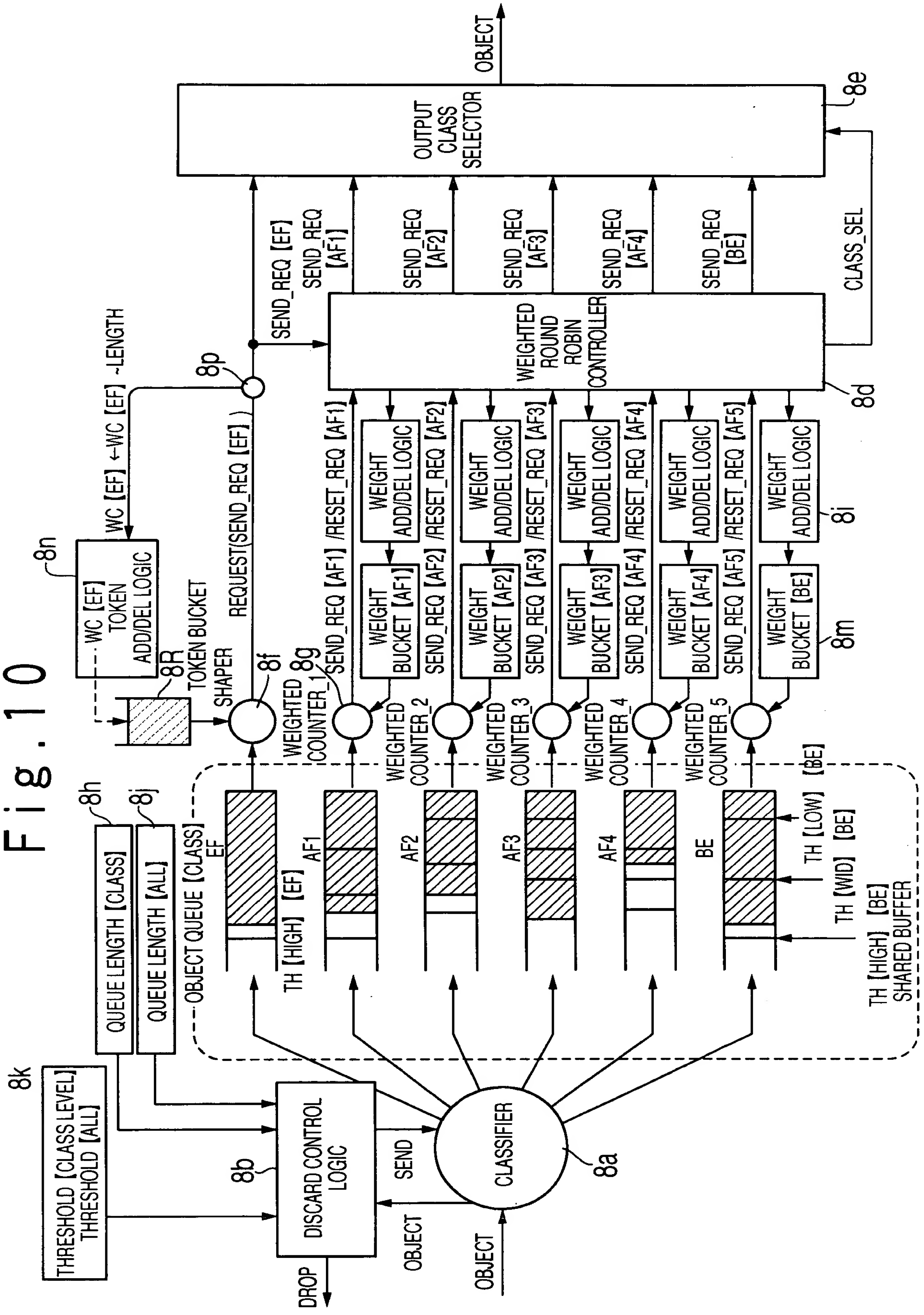

OBJECT
DISCARD CONTROL LOGIC
8b
DROP
OBJECT
SEND
CLASSIFIER
8a
THRESHOLD [CLASS LEVEL]
THRESHOLD [ALL]
8k
QUEUE LENGTH [CLASS]
8h
QUEUE LENGTH [ALL]
8j
OBJECT QUEUE [CLASS]
EF
AF1
AF2
AF3
AF4
BE
TH [HIGH] [EF]
TH [LOW] [BE]
TH [WID] [BE]
TH [HIGH] [BE]
SHARED BUFFER
8R
TOKEN BUCKET SHAPER
WC [EF] TOKEN ADD/DEL LOGIC
8n
WC [EF] ←WC [EF] ~LENGTH
8p
8f
REQUEST(SEND_REQ [EF])
WEIGHTED COUNTER_1
8g
WEIGHTED COUNTER_2
WEIGHTED COUNTER_3
WEIGHTED COUNTER_4
WEIGHTED COUNTER_5
SEND_REQ [AF1]
SEND_REQ [AF2]
SEND_REQ [AF3]
SEND_REQ [AF4]
SEND_REQ [AF5]
WEIGHT BUCKET [AF1]
WEIGHT BUCKET [AF2]
WEIGHT BUCKET [AF3]
WEIGHT BUCKET [AF4]
WEIGHT BUCKET [BE]
8m
WEIGHT ADD/DEL LOGIC
8i
SEND_REQ [AF1] /RESET_REQ [AF1]
SEND_REQ [AF2] /RESET_REQ [AF2]
SEND_REQ [AF3] /RESET_REQ [AF3]
SEND_REQ [AF4] /RESET_REQ [AF4]
SEND_REQ [AF5] /RESET_REQ [AF5]
WEIGHTED ROUND ROBIN CONTROLLER
8d
SEND_REQ [EF]
SEND_REQ [AF1]
SEND_REQ [AF2]
SEND_REQ [AF3]
SEND_REQ [AF4]
SEND_REQ [BE]
CLASS_SEL
OUTPUT CLASS SELECTOR
8e
OBJECT

Fig.11

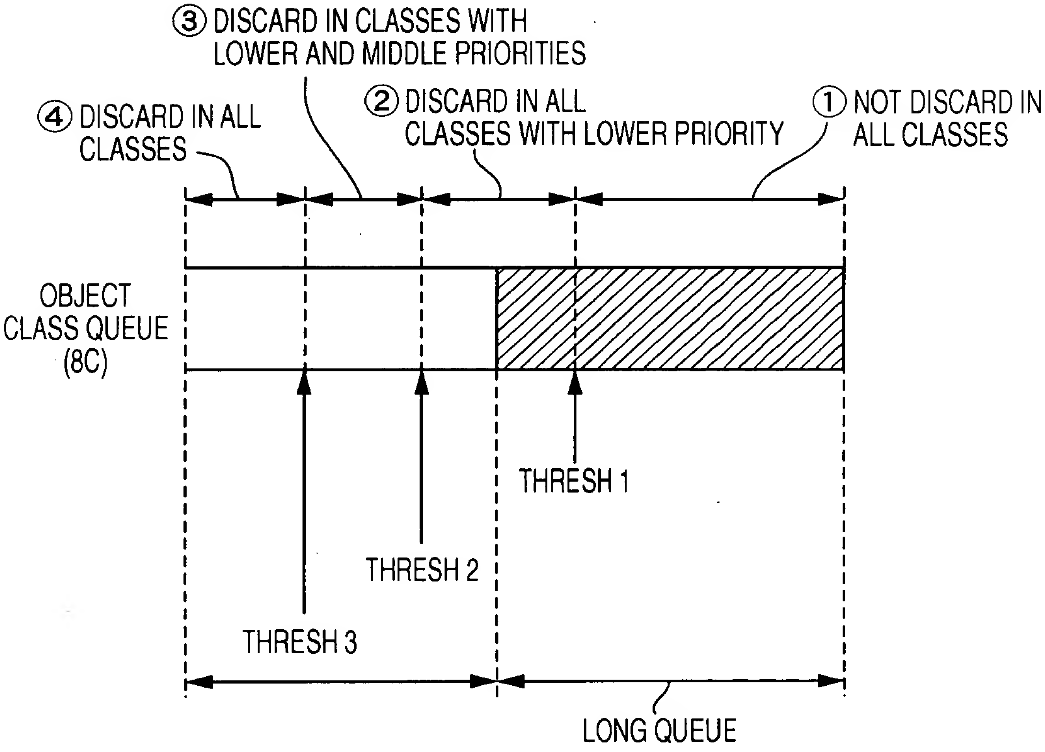

③ DISCARD IN CLASSES WITH LOWER AND MIDDLE PRIORITIES
④ DISCARD IN ALL CLASSES
② DISCARD IN ALL CLASSES WITH LOWER PRIORITY
① NOT DISCARD IN ALL CLASSES
OBJECT CLASS QUEUE (8C)
THRESH 1
THRESH 2
THRESH 3
LONG QUEUE

Fig.12

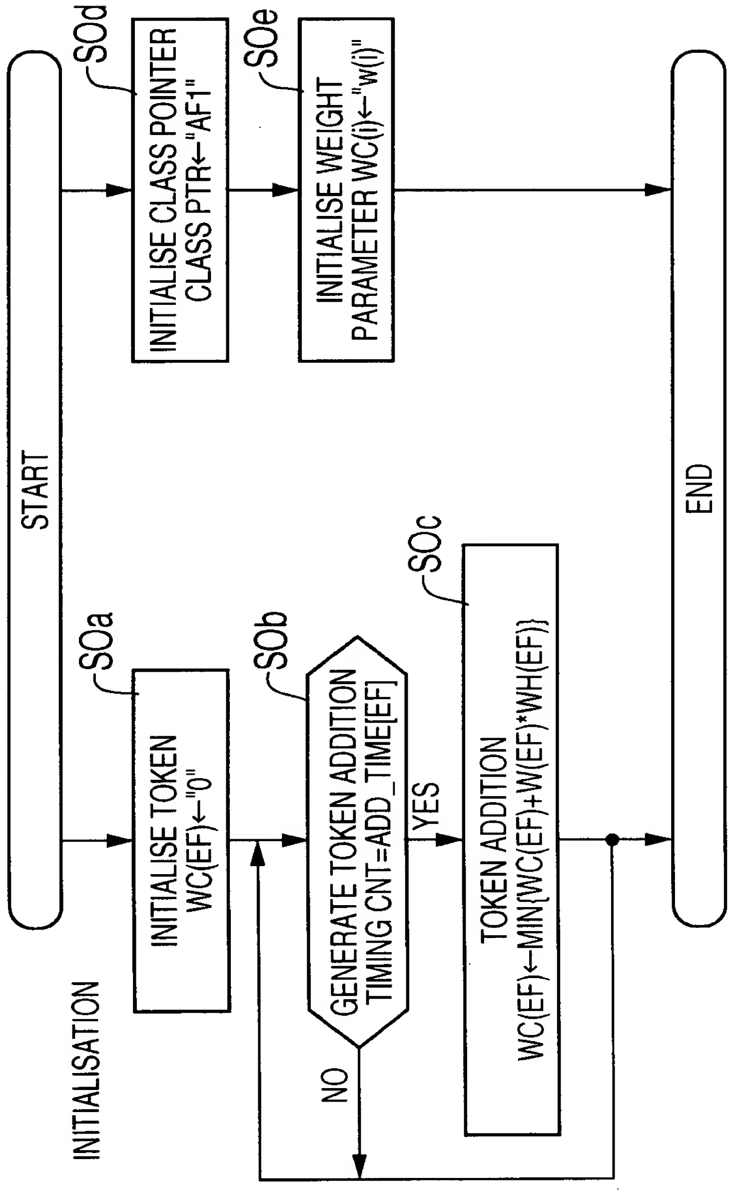

START
INITIALISATION
INITIALISE TOKEN WC(EF)←"0"
SOa
GENERATE TOKEN ADDITION TIMING CNT=ADD_TIME[EF]
SOb
YES
NO
TOKEN ADDITION WC(EF)←MIN{WC(EF)+W(EF)*WH(EF)}
SOc
INITIALISE CLASS POINTER CLASS PTR←"AF1"
SOd
INITIALISE WEIGHT PARAMETER WC(i)←"w(i)"
SOe
END

Fig. 13

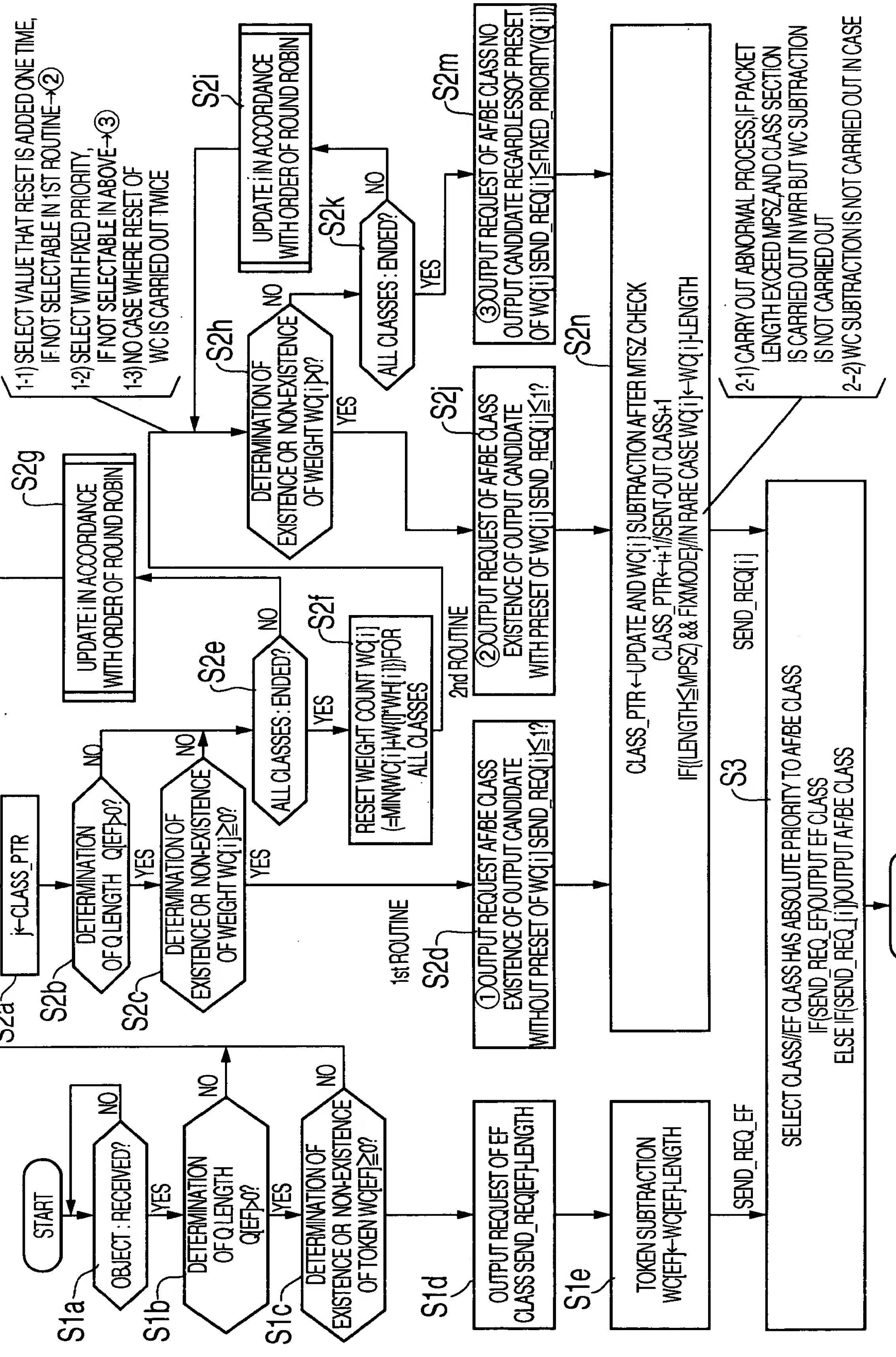

START
S1a
OBJECT : RECEIVED?
NO
YES
S1b
DETERMINATION OF Q LENGTH Q[EF]>0?
NO
YES
S1c
DETERMINATION OF EXISTENCE OR NON-EXISTENCE OF TOKEN WC[EF]≧0?
NO
S1d
OUTPUT REQUEST OF EF CLASS SEND_REQ[EF]-LENGTH
S1e
TOKEN SUBTRACTION WC[EF]←WC[EF]-LENGTH
SEND_REQ_EF
S2a
j←CLASS_PTR
S2b
DETERMINATION OF Q LENGTH Q[EF]>0?
NO
YES
S2c
DETERMINATION OF EXISTENCE OR NON-EXISTENCE OF WEIGHT WC[i]≧0?
NO
YES
1st ROUTINE
S2d
①OUTPUT REQUEST AF/BE CLASS EXISTENCE OF OUTPUT CANDIDATE WITHOUT PRESET OF WC[i] SEND_REQ[i]≦1?
S2e
ALL CLASSES : ENDED?
NO
YES
S2f
RESET WEIGHT COUNT WC[i] (=MIN(WC[i]+W[i]*WH[i]))FOR ALL CLASSES
S2g
UPDATE i IN ACCORDANCE WITH ORDER OF ROUND ROBIN
2nd ROUTINE
S2h
DETERMINATION OF EXISTENCE OR NON-EXISTENCE OF WEIGHT WC[i]>0?
NO
YES
S2j
②OUTPUT REQUEST OF AF/BE CLASS EXISTENCE OF OUTPUT CANDIDATE WITH PRESET OF WC[i] SEND_REQ[i]≦1?
S2k
ALL CLASSES : ENDED?
NO
YES
S2i
UPDATE i IN ACCORDANCE WITH ORDER OF ROUND ROBIN
S2m
③OUTPUT REQUEST OF AF/BE CLASS NO OUTPUT CANDIDATE REGARDLESS OF PRESET OF WC[i] SEND_REQ[i]≦FIXED_PRIORITY(Q[i])
1-1) SELECT VALUE THAT RESET IS ADDED ONE TIME, IF NOT SELECTABLE IN 1ST ROUTINE→②
1-2) SELECT WITH FIXED PRIORITY, IF NOT SELECTABLE IN ABOVE→③
1-3) NO CASE WHERE RESET OF WC IS CARRIED OUT TWICE
S2n
CLASS_PTR←UPDATE AND WC[i] SUBTRACTION AFTER MTSZ CHECK
CLASS_PTR←i+1/SENT-OUT CLASS+1
IF((LENGTH≦MPSZ) && FIXMODE)/IN RARE CASE WC[i]←WC[i]-LENGTH
SEND_REQ[i]
2-1) CARRY OUT ABNORMAL PROCESS, IF PACKET LENGTH EXCEED MPSZ, AND CLASS SECTION IS CARRIED OUT IN WRR BUT WC SUBTRACTION IS NOT CARRIED OUT
2-2) WC SUBTRACTION IS NOT CARRIED OUT IN CASE
S3
SELECT CLASS//EF CLASS HAS ABSOLUTE PRIORITY TO AF/BE CLASS
IF(SEND_REQ_EF)OUTPUT EF CLASS
ELSE IF(SEND_REQ_[i])OUTPUT AF/BE CLASS
END

Fig. 14

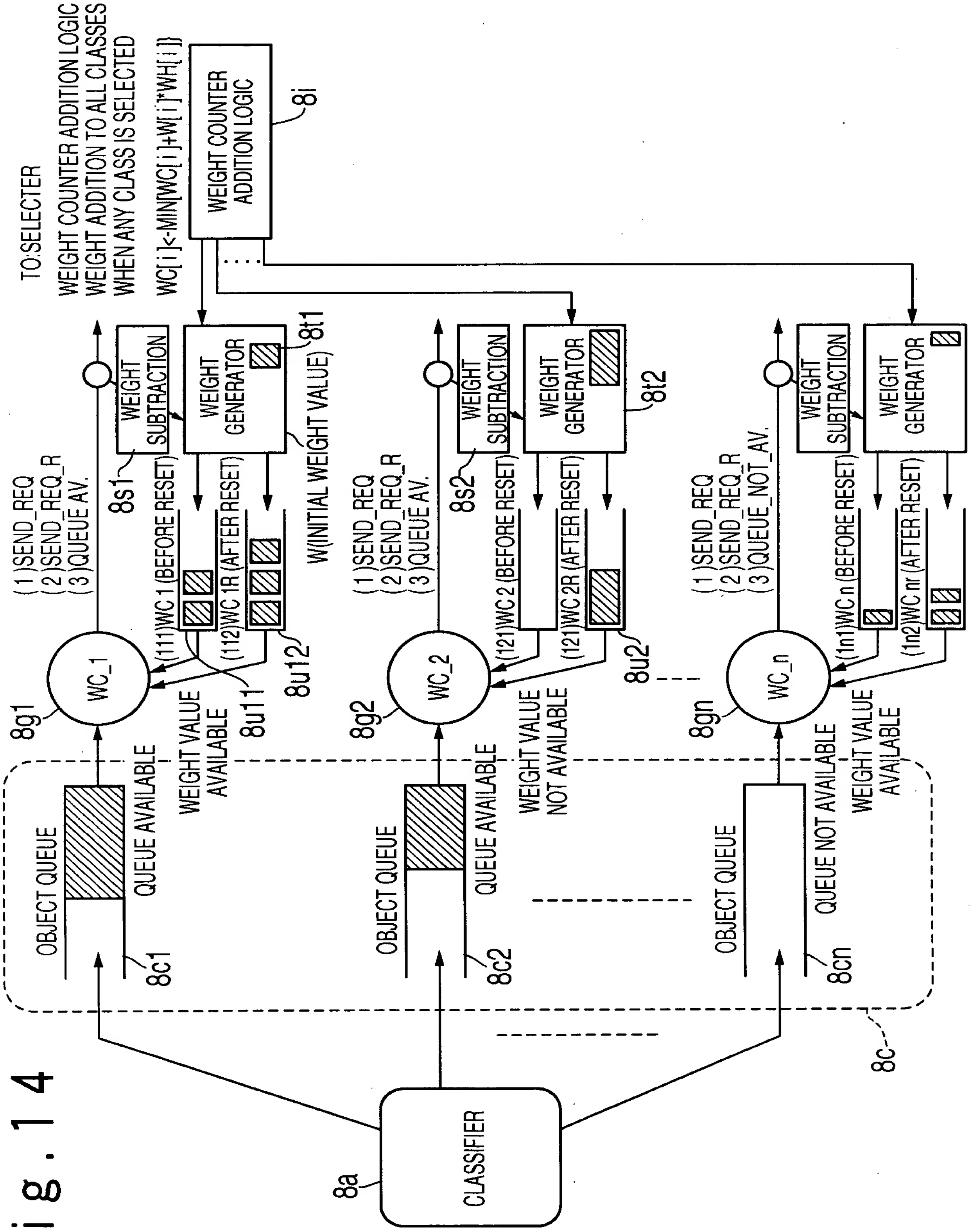

TO:SELECTER
WEIGHT COUNTER ADDITION LOGIC
WEIGHT ADDITION TO ALL CLASSES
WHEN ANY CLASS IS SELECTED
WC[i]<-MIN[WC[i]+W[i]*WH[i]]
WEIGHT COUNTER ADDITION LOGIC
8i
(1)SEND_REQ
(2)SEND_REQ_R
(3)QUEUE AV.
8s1
WEIGHT SUBTRACTION
WEIGHT GENERATOR
8t1
W(INITIAL WEIGHT VALUE)
(111)WC1 (BEFORE RESET)
(112)WC1R (AFTER RESET)
8u11
8u12
8g1
WC_1
WEIGHT VALUE AVAILABLE
OBJECT QUEUE
QUEUE AVAILABLE
8c1
(1)SEND_REQ
(2)SEND_REQ_R
(3)QUEUE AV.
8s2
WEIGHT SUBTRACTION
WEIGHT GENERATOR
8t2
(121)WC2 (BEFORE RESET)
(121)WC2R (AFTER RESET)
8u2
8g2
WC_2
WEIGHT VALUE NOT AVAILABLE
OBJECT QUEUE
QUEUE AVAILABLE
8c2
(1)SEND_REQ
(2)SEND_REQ_R
(3)QUEUE_NOT_AV.
WEIGHT SUBTRACTION
WEIGHT GENERATOR
(1n1)WCn (BEFORE RESET)
(1n2)WCnr (AFTER RESET)
8gn
WC_n
QUEUE NOT AVAILABLE
WEIGHT VALUE AVAILABLE
OBJECT QUEUE
8cn
8c
CLASSIFIER
8a

Fig. 15

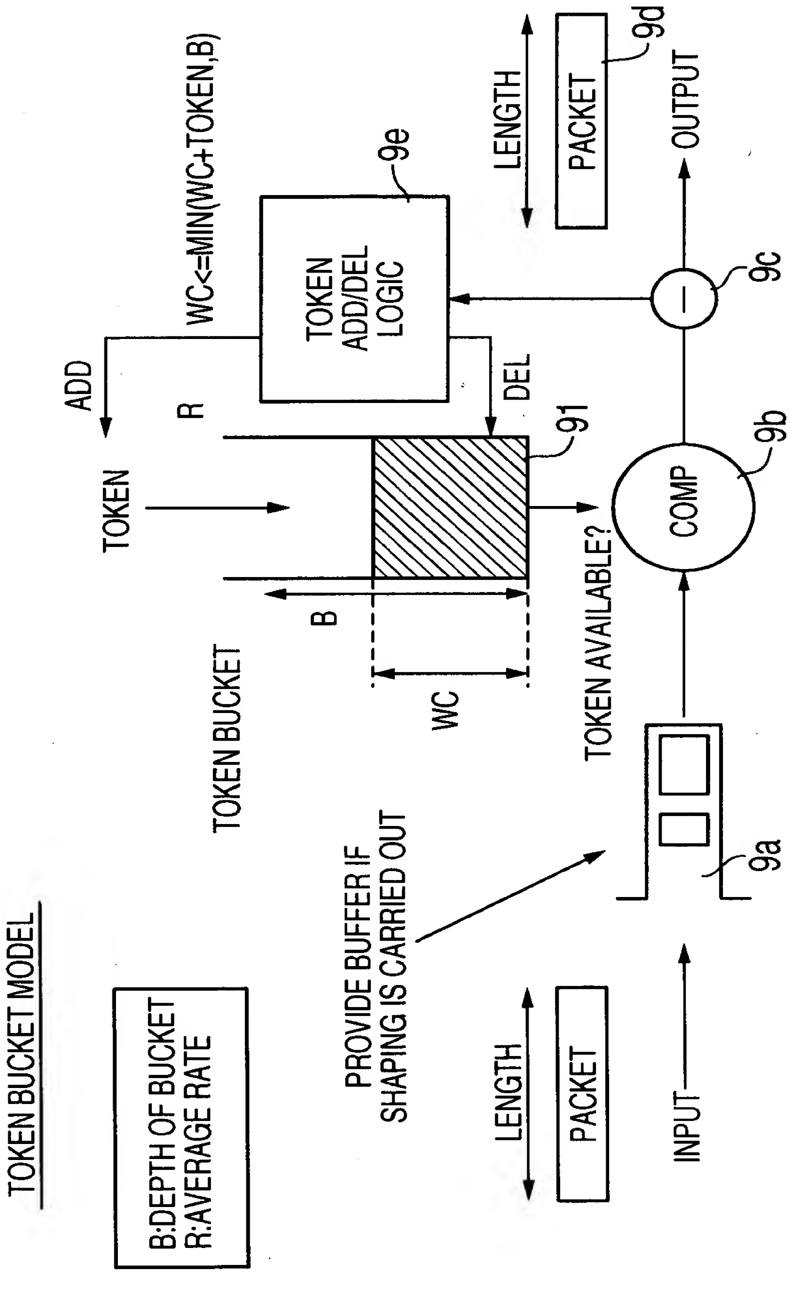
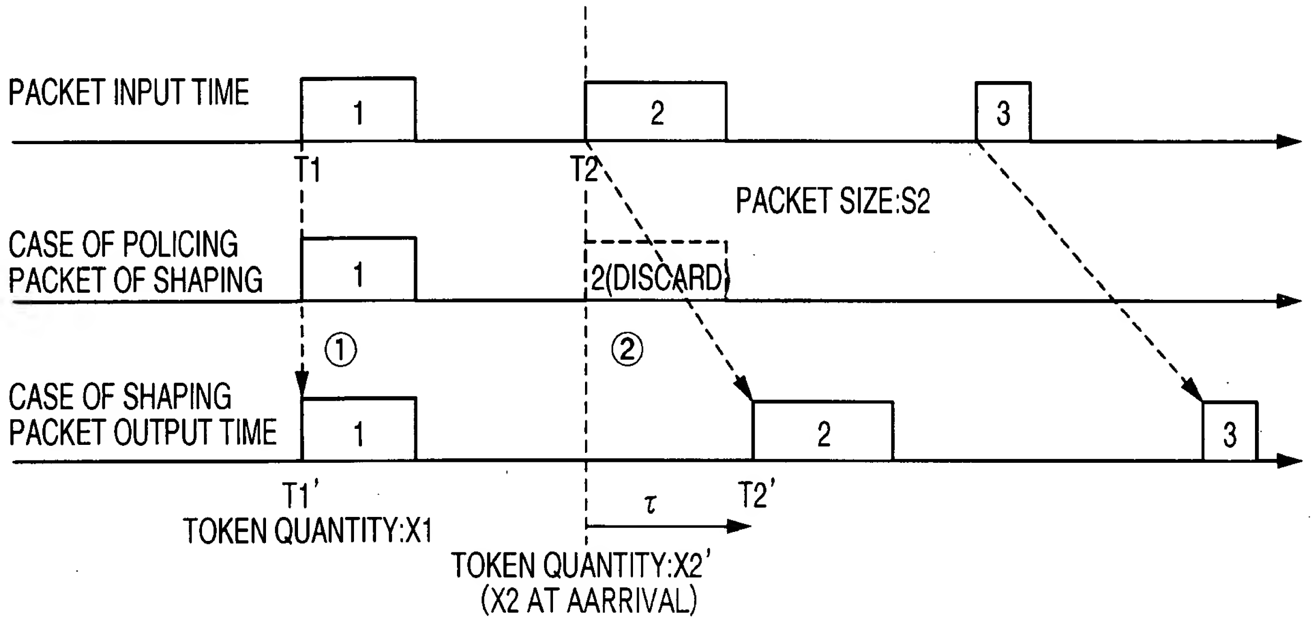

TOKEN BUCKET MODEL
B:DEPTH OF BUCKET
R:AVERAGE RATE
PROVIDE BUFFER IF SHAPING IS CARRIED OUT
LENGTH
PACKET
INPUT
9a
TOKEN BUCKET
WC
B
TOKEN AVAILABLE?
TOKEN
ADD
R
91
COMP
9b
DEL
TOKEN ADD/DEL LOGIC
9e
WC<=MIN(WC+TOKEN,B)
9c
LENGTH
PACKET
9d
OUTPUT

Fig.16



1)TOKEN QUANTITIY AT TIME T2:X2=x1+(T2-T1)*R
LACK OF TOKEN,IF S2>X2
2)POLICING
IMMEDIATELY DISCARD
3)SHAPING
NOT LACK OF TOKEN,IF PACKET IS TRANSMITTED AT TIME
(τ+T2)(S2=X1+{(τ+T2)-T1}*R).PACKET IS TRANSMITTED WITH DELAY τ